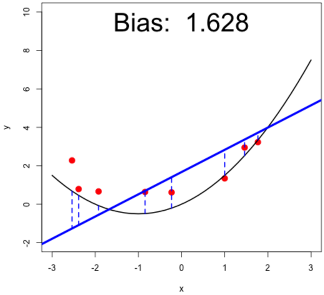
Introduction to Machine Learning

Deep Dive: Bias-Variance Decomposition



Learning goals

- Understand how to decompose the generalization error of a learner into
  - Bias of the learner
  - Variance of the learner
  - Inherent noise in the data

BIAS-VARIANCE DECOMPOSITION

Let us take a closer look at the generalization error of a learning algorithm $\mathcal{I}_L$. This is the expected error of an induced model $\hat{f}_{\mathcal{D}_n}$, on training sets of size n, when applied to a fresh, random test observation.

$$GE_n\left(\mathcal{I}_L\right) = \mathbb{E}_{\mathcal{D}_n \sim \mathbb{P}_{xy}^n, (\mathbf{x}, y) \sim \mathbb{P}_{xy}} \left(L\left(y, \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right) = \mathbb{E}_{\mathcal{D}_n, xy} \left(L\left(y, \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)$$

We therefore need to take the expectation over all training sets of size n, as well as the independent test observation.

We assume that the data is generated by

$$y = f_{\text{true}}(\mathbf{x}) + \epsilon\,,$$

with zero-mean homoskedastic error $\epsilon \sim (0, \sigma^2)$ independent of **x**.

BIAS-VARIANCE DECOMPOSITION

By plugging in the $L2$ loss $L(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$ we get

$$
\begin{aligned}
GE_n\left(\mathcal{I}_L\right) &= \mathbb{E}_{\mathcal{D}_n, xy}\left(L\left(y, \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right) = \mathbb{E}_{\mathcal{D}_n, xy}\left(\left(y - \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)^2\right) \\
&\overset{\text{LIE}}{=} \mathbb{E}_{xy}\Big[\underbrace{\mathbb{E}_{\mathcal{D}_n}\left(\left(y - \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)^2 \mid \mathbf{x}, y\right)}_{(*)}\Big]
\end{aligned}
$$

Let us consider the error $(*)$ conditioned on one fixed test observation $(\mathbf{x}, y)$ first. (We omit the $\mid \mathbf{x}, y$ for better readability for now.)

$$
\begin{aligned}
(*) &= \mathbb{E}_{\mathcal{D}_n}\left(\left(y - \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)^2\right) \\
&= \underbrace{\mathbb{E}_{\mathcal{D}_n}\left(y^2\right)}_{=y^2} + \underbrace{\mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})^2\right)}_{(1)} - 2\underbrace{\mathbb{E}_{\mathcal{D}_n}\left(y\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)}_{(2)}
\end{aligned}
$$

by using the linearity of the expectation.

BIAS-VARIANCE DECOMPOSITION

$$(*) = \mathbb{E}_{\mathcal{D}_n}\left(\left(y - \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)^2\right) = y^2 + \underbrace{\mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})^2\right)}_{(1)} - 2\,\underbrace{\mathbb{E}_{\mathcal{D}_n}\left(y\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)}_{(2)} =$$

Using that $\mathbb{E}(z^2) = \text{Var}(z) + \mathbb{E}^2(z)$, we see that

$$= y^2 + \text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) + \mathbb{E}^2_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) - 2y\mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x}))\right)$$

Plug in the definition of y

$$= f_{\text{true}}(\mathbf{x})^2 + 2\epsilon f_{\text{true}}(\mathbf{x}) + \epsilon^2 + \text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) + \mathbb{E}^2_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) - 2(f_{\text{true}}(\mathbf{x}) + \epsilon)\mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x}))\right)$$

Reorder terms and use the binomial formula

$$= \epsilon^2 + \text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) + \left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)^2 + 2\epsilon\left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)$$

BIAS-VARIANCE DECOMPOSITION

$$(*) = \epsilon^2 + \text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) + \left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)^2 + 2\epsilon\left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)$$

Let us come back to the generalization error by taking the expectation over all fresh test observations $(\mathbf{x}, y) \sim \mathbb{P}_{xy}$:

$$\begin{aligned} GE_n(\mathcal{I}_L) &= \underbrace{\sigma^2}_{\text{Variance of the data}} + \mathbb{E}_{xy}\underbrace{\left[\text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x}) \mid \mathbf{x}, y\right)\right]}_{\text{Variance of learner at } (\mathbf{x}, y)} \\ &+ \mathbb{E}_{xy}\underbrace{\left[\left(\left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)^2 \mid \mathbf{x}, y\right)\right]}_{\text{Squared bias of learner at } (\mathbf{x}, y)} + \underbrace{0}_{\text{As } \epsilon \text{ is zero-mean and independent}} \end{aligned}$$

BIAS-VARIANCE DECOMPOSITION

$GE_n(\mathcal{I}_L) =$

$$\underbrace{\sigma^2}_{\text{Variance of the data}} + \mathbb{E}_{xy}\underbrace{\left[\mathrm{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x}) \mid \mathbf{x}, y\right)\right]}_{\text{Variance of learner at } (\mathbf{x}, y)} + \mathbb{E}_{xy}\underbrace{\left[\left(\left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)^2 \mid \mathbf{x}, y\right)\right]}_{\text{Squared bias of learner at } (\mathbf{x}, y)}$$

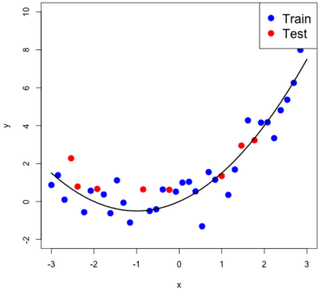
1. The first term expresses the variance of the data. This is pure **noise** in the data. Also called Bayes, intrinsic or irreducible error. No matter what we do, we will never get below this error.
2. The second term expresses, on average, how much $\hat{f}_{\mathcal{D}_n}(\mathbf{x})$ fluctuates around test points if we vary the training data. Expresses also the learner's tendency to learn random things irrespective of the real signal (overfitting).
3. The third term says how much we are "off" on average at test locations (underfitting). Models with high capacity typically have low **bias** and *vice versa*.

BIAS-VARIANCE DECOMPOSITION

Illustration: Let us consider the following example. We will generate a dataset using the following model :

$$y = x + \frac{x^2}{2} + \epsilon\,, \quad \epsilon \sim N(0,1)$$

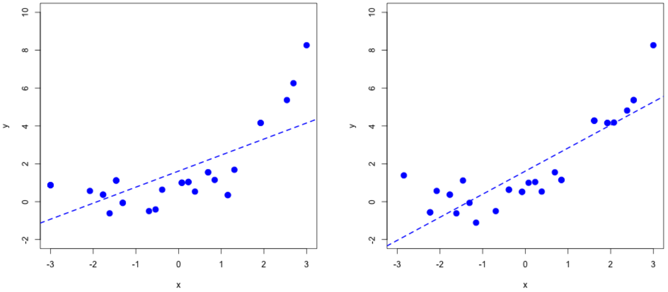
The data is then split into a training set and a test set.



BIAS-VARIANCE DECOMPOSITION

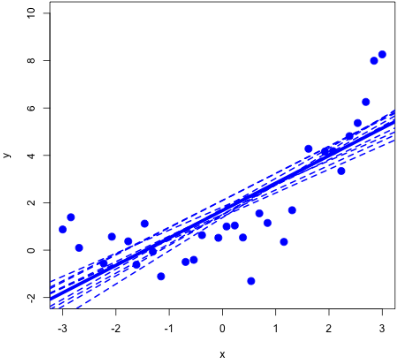
To obtain estimates for the bias and variance, we will train several models by sampling with replacement from the training data. This is commonly known as **bootstrapping**.

First, we train several (low capacity) linear models (polynomial of degree $d = 1$).



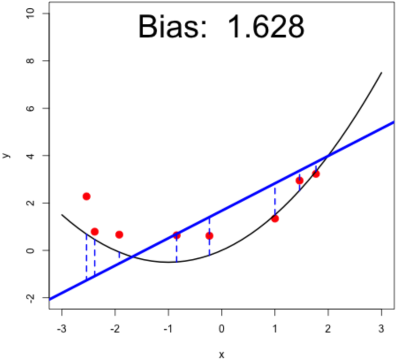
BIAS-VARIANCE DECOMPOSITION

By creating several models, we obtain the average model over different samples of the training dataset.



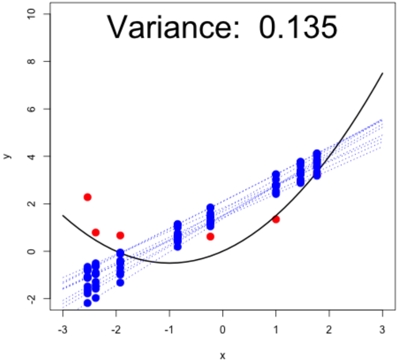
BIAS-VARIANCE DECOMPOSITION

We can now estimate the (squared) bias, by computing the average squared difference between the average model and the true model, at the test point locations.



BIAS-VARIANCE DECOMPOSITION

We compute the average variance of the predictions of the models we trained at the test point locations.

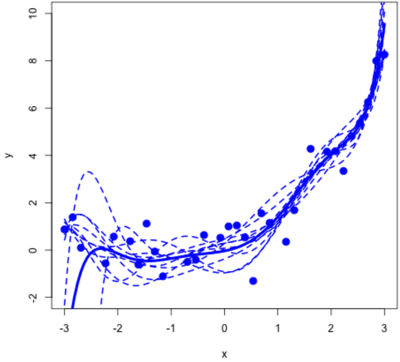


$$GE_n(\mathcal{I}_L) \approx 1 + 1.628 + 0.135 = 2.763$$

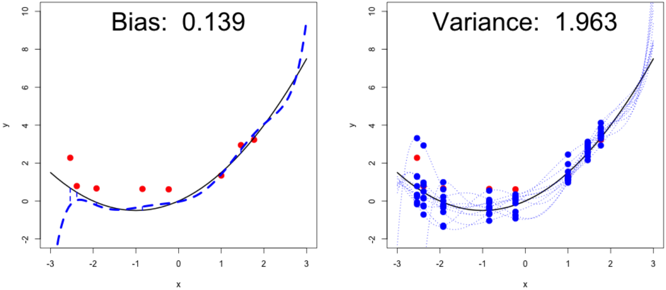
- The biggest component of the generalization error is the bias.

BIAS-VARIANCE DECOMPOSITION

We will repeat the same procedure, but use a high-degree polynomial ($d = 7$) with more capacity.



BIAS-VARIANCE DECOMPOSITION

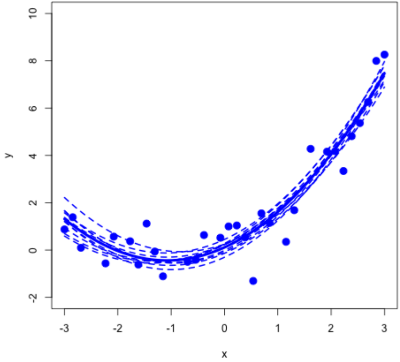


$$GE_n(\mathcal{I}_L) \approx 1 + 0.139 + 1.963 = 3.102$$

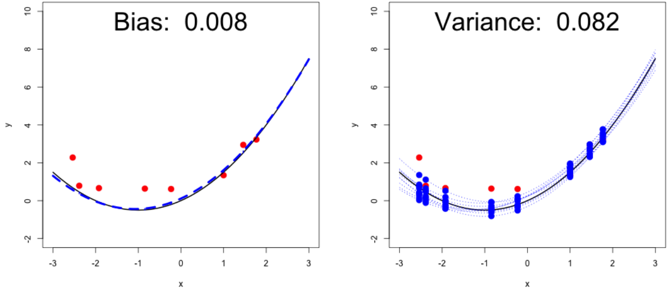
- The generalization error is higher than before
- Even though the bias is lower, the variance of the learner is higher.

BIAS-VARIANCE DECOMPOSITION

What happens if we use a model with the same complexity as the true model (quadratic polynomial)?



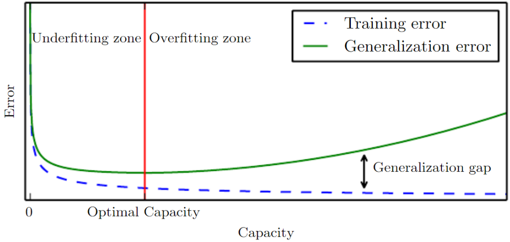
BIAS-VARIANCE DECOMPOSITION



$$GE_n\left(\mathcal{I}_L\right) \approx 1 + 0.008 + 0.082 = 1.091$$

- The generalization error is the lowest at this complexity.
- The variance of the data acts as a lower bound.

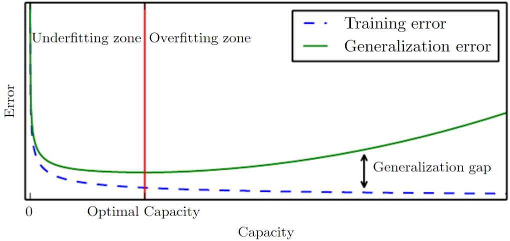
CAPACITY AND OVERFITTING



Credit: Ian Goodfellow

- The performance of a learner depends on its ability to
  1. **fit** the training data well
  2. **generalize** to new data
- Failure of the first point is called **underfitting**
- Failure of the second item is called **overfitting**

CAPACITY AND OVERFITTING



Credit: Ian Goodfellow

- The tendency of a model to underfit/overfit is a function of its capacity, determined by the type of hypotheses it can learn.
- The generalization error is minimized when it has the right capacity.
- Even for correctly specified models, the generalization error is lower-bounded by the irreducible noise σ^2.